PTO-103P
(Rev. 8-95) PROVISIONAL APPLICATION
FILING RECEIPT

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office
ASSISTANT SECRETARY AND COMMISSIONER
OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

APPLICATION NUMBER	FILING DATE	FIL FEE REC'D	ATTORNEY DOCKET NO.	DRWGS
60/101,851	09/25/98	$75.00	A-311	9

DELLETT AND WALTERS
310 S W FOURTH AVENUE
SUITE 1101
PORTLAND OR 97204

Receipt is acknowledged of this Provisional Application. This Provisional Application will not be examined for patentability. Be sure to provide the PROVISIONAL APPLICATION NUMBER, FILING DATE, NAME OF APPLICANT, and TITLE OF INVENTION when inquiring about this application. Fees transmitted by check or draft are subject to collection. Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please write to Box Provisional Application within 10 days of receipt. Please provide a copy of the Provisional Application Filing Receipt with the changes noted thereon. This Provisional Application will automatically be abandoned twelve (12) months after its filing date and will not be subject to revival to restore it to pending status beyond a date which is after twelve (12) months from its filing date.

Applicant(s) KENNETH L. LEVY, STEVENSON, WA.

FOREIGN FILING LICENSE GRANTED 10/19/98 * SMALL ENTITY *

TITLE
METHOD AND APPARATUS FOR EMBEDDING AUXILIARY INFORMATION WITHIN ORIGINAL DATA

RECEIVED
OCT 23 1998

DELLETT AND WALTERS

METHOD AND APPARATUS FOR EMBEDDING AUXILIARY INFORMATION WITHIN ORIGINAL DATA

Background of the Invention

This invention relates to the field of signal processing, and more specifically, to techniques for hiding auxiliary information within original data.

With the recent explosive growth in the use of electronic information, enforcement of copyright laws has become more difficult. The cost of the equipment required to copy digital data representing music, art, and other valuable information has been decreasing, while the capacity of readily available data storage media has been increasing. Inexpensive devices can write enormous amounts of data to digital storage media such as writable compact disks (CDs), multi-gigabyte hard disk drives, high capacity removable magnetic disks, and soon to be available digital versatile disks (DVDs). Readily available high resolution printers and scanners bring the digitization and reproduction of graphic information within the means of most consumers. In addition, readily available high resolution sound cards, including analog-to-digital and digital-to-analog converters, bring the digitization and reproduction of audio information within the means of most consumers. Not only is copying digital files simple and inexpensive, the Internet facilitates unauthorized distribution of copyrighted works.

Unlike analog copies, which are always inferior to the original, a copy of digital information can be identical to that of the original, with no degradation due to copying. Millions of dollars are lost annually due to illegal but exact duplications of digital media and near-exact duplications of analog media. Because copying equipment is readily available, catching

persons making unauthorized copies can be difficult. Even if an unauthorized copier is apprehended, the creator of the original document must still prove that the allegedly unauthorized copy was in fact copied from his original work and not independently created.

One approach to solving the problem of illegal copying is embedding or hiding authentication information or copy protection information within the original data. Hiding auxiliary information in original data, also called steganography, has been used for thousands of years. In steganography, a message is hidden within another object or media, so that the message is not perceived by an observer. Steganography is related to, but different from, cryptography, in which the existence of a message is typically obvious, but its meaning is not ascertainable without special knowledge.

Hidden data, also referred to as auxiliary or embedded data, can be used to prevent unauthorized copying by embedding in the original data commands that are readable by the copying device and that instruct the copying device not to make a usable copy. Hidden data can also be used to authenticate data, that is, to prove authorship. One such technique entails embedding auxiliary information in an original work in such a manner that special knowledge, such as a secret algorithm or code, is required to detect and/or remove the auxiliary information. The copier would not be able to remove the authentication information, and the original creator could prove his authorship by retrieving the embedded information, which would identify him as the author.

Data hiding has uses besides the prevention and detection of unauthorized copying. One such use is content enhancement, that is, adding information to the original data to enhance the content. For example,

lyrics could be embedded in audio data on a CD. The lyrics could be viewed in a special kareoke machine, while the audio could be played on an existing CD player. Hidden data could also be used to associate different segments of video data with different viewer-selectable versions of the video on a DVD. For example, a viewer could select between a version edited for children or an unabridged version, and embedded auxiliary data would indicate to the DVD player which video segments to skip and which to include for the selected version.

The original data in which the auxiliary data is hidden may represent any type of information that is perceivable with the aid of a presenting device. For example, the data may represent music which is presented using a compact disk or audio DVD player, a video film that is presented on a DVD player, or an image that is presented on a computer screen or a printer.

When the combined data is presented to a user by a normal presentation device, the auxiliary data should not interfere with the use of the original data. Ideally, the user should not be able to perceive the auxiliary data at all. Unfortunately, increasing the amount of the embedded auxiliary data or its robustness, that is, its persistence to attack and data transformation, may incidentally increase its perceptibility. The degree to which the auxiliary data can be perceived without having an adverse impact on the user varies with the application. For example, in CD quality audio, a minor change from the original data might result in unacceptable audio artifacts. In video data, a minor change in a presented image may be acceptable, even though the change might be noticeable if the original and combined works are presented and compared side by side.

Several techniques are known for hiding auxiliary information in original digital data. Data can be hidden in original data as headers or trailers appended to the original data. Such techniques are of limited use in protection of copyrighted works, because the auxiliary data is easily located and stripped out of the copy, as when changing format. More sophisticated techniques distribute the auxiliary data through the original data, entwining the auxiliary and original data until the auxiliary data is difficult, or even statistically impossible, to identify and strip from the combined data.

Most data hiding techniques that distribute the auxiliary data through the original data are computationally intense and therefore expensive to implement. Many of these techniques are based upon adding or subtracting periods of pseudo-random noise (PN) sequences with the signal to represent the auxiliary information. The rest are based upon adding the auxiliary information to the original data after the original data has been transformed into the frequency domain, such as by a Fourier transform. Auxiliary information can be added in the frequency domain so that the energy of the auxiliary data is spread across many frequencies in a manner similar to that of the PN sequence. In addition, auxiliary information can be added to the phase of the frequency components with and without spreading the information across frequencies. Unfortunately, transforming the data into the frequency domain and/or shaping the energy of the PN sequence so it is non-perceivable requires intense calculations.

The ability of users to detect auxiliary data depends not only upon the data, but also upon the characteristics of the human senses organs and the

interpretation of sensory stimuli by the brain. Some data hiding techniques transform the original data into the frequency domain and embed auxiliary data in a manner such that the frequency spectrum of the original data reduces the perception of embedded data. This psychophysical effect is known as masking. The frequency distribution of the original data is used to determine preferred frequencies at which the embedded auxiliary data will be less perceptible, that is, masked. Others use the fact that we don't perceive phase as accurately as magnitude in the frequency domain.

There are some data embedding techniques that are less computationally intense and that still distribute the auxiliary data in the original data. Such techniques include amplitude modulation, frequency band elimination, distinct quantization, and least-significant bit (LSB) replacement. These techniques embed data in predetermined locations without regard to the original data and are, therefore, more likely to produce perceptual side affects in the combined data. In addition, the LSB replacement technique is easily disturbed by low level noise.

The ease of retrieving embedded data varies with the technique used for embedding. Some data hiding and retrieving techniques retrieve the auxiliary data by comparing the combined data with the original data. Others retrieve the auxiliary information using databases of the PN sequences that were originally used to hide the data. Techniques that require that a copy of the original data or a PN database be used to extract the auxiliary data are of limited use in applications in which the combined data is distributed broadly. Such techniques are useful in some applications, such as data authentication, in which the

auxiliary data is retrieved rarely and only by the copyright owner.

Thus, it would be desirable to have a data hiding and extracting technique that is not computationally intense and that maintains a desired level of imperceptibility and robustness, and in which the embedded data that can be retrieved from the combined data without access to the original data.

Summary of the Invention

Accordingly, it is an object of the present invention to provide a method and apparatus of data hiding and retrieval.

An advantage of the present invention is its extreme efficiency relative to the prior art, that is, the computations involved in embedding and retrieval data according to the invention are not intense, so that data retrieval can be accomplished by a normal data reading device, such as a CD or DVD player, with little or no slowing of the reading device, and data can be readily embedded in a large number of files without having to invest in elaborate equipment.

Another advantage of the invention is that it uses an adaptive algorithm based upon psychophysical masking, that is, the invention uses the original data, in unmodified form or without complex data transformations, to determine the location to insert auxiliary data, and, in many embodiments, the invention uses the value of the original data to determine the value to set the data point that contains the embedded data, thereby increasing the ease of data embedding while minimizing or eliminating user perception of the auxiliary data.

A further advantage of the invention is that the embedded data can be imperceptible when presented with

existing players.

Still another advantage of the invention is that it uses a broadband approach that distributes the auxiliary data through the original data, thereby making the auxiliary data more difficult to detect and remove.

Yet another advantage of the invention is that it provides for a minimal decrease in signal-to-noise ratio (SNR) of the combined data as compared to the original data.

Still a further advantage of the invention is that it provides non-LSB capabilities, thereby providing increased robustness, that is, the auxiliary data is unchanged by some transformations of the combined data, such as by conversions between the analog and digital domains.

Yet a further advantage of the invention is that it can cause no detectable change in data's statistics, thereby making authentication information impossible to identify and remove.

Another advantage of the invention is that it does not require the original file or other database for extracting the auxiliary data.

Still another advantage of the invention is its versatility, in that it allows the user to set parameters that vary the perceptibility, robustness, and embedding rate so that the invention can be used in a broad variety of applications.

The present invention involves encoding and decoding auxiliary information into original data to produce combined data. The invention uses a detection criterion or criteria to determine where in the original data to locate and adjust data points to carry the auxiliary information. The detection criteria is used to locate positions, referred to as local masking

opportunities, in the original data at which the embedding of auxiliary data will produce minimal perception based upon the psychophysical masking.

The data points in the original data are investigated in accordance with the detection criteria to determine the existence of a local masking opportunities. The detection criterion or criteria may involve, for example, comparing the data point to a predetermined value and examining the relationship of the data point to nearby points. If the detection criteria are met, one or more of the nearby points, or the data point being investigated, is changed to indicate the value of an embedded bit of auxiliary data.

Thus, although the search for local masking opportunities typically progresses point by point through the data, the investigation of each point may include not only the value of that point, but also values of one or more nearby points and/or one or more relationships among the points. If the investigation of a point shows the existence of a local masking opportunity, data is embedded by setting the value of one or more of the local points, that is, either the point being investigated or one or more of the nearby points

The value to which the nearby data points are set is typically dependent upon the data point being investigated, as well as on the value of the auxiliary data bit. The data point value can be set so that it has a specified relationship with the data point. The process is continued until the original data has been traversed or no additional auxiliary data remains to be embedded.

Extracting the auxiliary data is the inverse of the embedding process. The combined data is traversed

using the detection criteria to locate the local masking opportunities. As each local masking opportunity is located, the nearby data point or points that was or were set to indicate the embedded bit is or are read to extract the embedded data. The process is continued until the combined data has been traversed.

In the preferred embodiments, a data point or points are set to a value relative to the nearby data point and not to an absolute value. Setting data points at the local masking opportunity, and setting the data point to a value related to the nearby point, rather than to a value unrelated to the original data, both provide masking that reduces the perceptibility of the data. The data is extracted by determining the relationships or values of the point or points near the local masking opportunity.

For the two preferred embodiments described in detail below, only points with large values are adjusted, and by a minimal amount; thus, these embodiments are based upon the masking of a weak stimulus by an intense stimulus. The process is applicable to analog and digital data. However, both embodiments are explained in terms of digital media due to current switch to digital media and the ease of understanding.

Specifically, the first preferred embodiment uses the difference between a data point after a peak and the peak level to carry auxiliary information, as long as the peak is above a large threshold and the original difference between the peak and next point is not too great. This large threshold and minimal difference produces the desired perceptual masking. The encoding process adjusts the point after the above-threshold peaks to embed the data, whereas the decoding process measures the difference between each above threshold

peak level and the next data point to retrieve the auxiliary data.

The second preferred embodiment uses the change in slope across a positive, large, steep, threshold crossing to hide the auxiliary information, as long as the original change in slope is not too great yet steep enough to accept the ensuing adjustment. Again, the large threshold produces the desired perceptual masking. In the implementation, the encoding process adjusts the change in slope to embed the data, whereas the decoding process measures the change in slope to retrieve the auxiliary data.

The preferred embedding process implicitly spreads the energy of the auxiliary information throughout the original data. This broadband approach produces data that is more difficult to remove than sub-band approaches that place the data in an inaudible frequency range. If desired, parameters can be chosen so that the process produces protected data that is statistically identical to unmarked data. Importantly, the process can be adjusted to produce the desired tradeoffs between perception, coding rate and robustness to attack. Such an embodiment of the invention preferably operates on the original data without requiring any data transformations, such as a Fourier transformation. Compared to the prior data embedding techniques that use masking, this process is therefore extremely efficient, thus allowing it to be implemented at a much lower cost than competing techniques.

The invention is efficient because it operates on the original data without using computationally intense data transformations, such as transforming the data into the frequency domain. If the original data represents information in the time domain, the data can

remain in the time domain as the auxiliary data is embedded and retrieved. Of course, the invention can operate on original data of all types, and not just data in the time domain.

In summary, the present invention has the advantage of being extremely efficient to implement while still maintaining a desired level of robustness, i.e., resistance to tampering, including non-LSB (least significant bit) capabilities and statistical invisibility. The efficiency of the present invention means that it is cheaper to implement and can be used to encode, decode and search more files than a computationally intense algorithm. Further objects and advantages will become apparent from a consideration of the ensuing description and drawings.

The subject matter of the present invention is particularly pointed out and distinctly claimed in the concluding portion of this specification. However, both the organization and method of operation, together with further advantages and objects thereof, may best be understood by reference to the following description taken in connection with accompanying drawings wherein like reference characters refer to like elements.

Brief Description of the Drawings

FIG. 1 is a flowchart showing in general the steps embedding data using the invention;

FIG. 2 is a block diagram showing an apparatus used to embed or extract data using the process of FIG. 1;

FIG. 3 is a flowchart showing in general the steps used to extract the data embedded using the process of FIG. 1;

FIG. 4 is a block diagram showing conceptually the operation of a first embodiment of the invention;

FIG. 5 is a flowchart showing the steps of encoding data in accordance with the first embodiment of the invention;

FIG. 6 is a flowchart showing the steps of extracting data that was encoded in accordance with the embodiment shown in FIG. 5;

FIG. 7 is a block diagram showing conceptually the operation of a second embodiment of the invention;

FIG. 8 is a flowchart showing the steps of encoding data in accordance with a second embodiment of the invention;

FIG. 9 is a flowchart showing the steps of extracting data that was encoded in accordance with the embodiment shown in FIG. 8;

FIG. 10 demonstrates the operation of the invention in conjunction with digital compression techniques;

FIG. 11 is a block diagram showing an apparatus used to carry out the invention;

FIG. 12 shows a first embodiment of the apparatus of FIG. 2; and

FIG. 13 is a block diagram showing a second embodiment of the apparatus of FIG. 2

Detailed Description

The system according to a preferred embodiment of the present invention comprises a method and apparatus for hiding auxiliary data in original data and for retrieving the auxiliary data.

FIG. 1 shows broadly the steps involved in carrying out a method of the invention to embed data. Fig. 2 shows a block diagram of an apparatus 10 used to perform the method of FIG. 1. Apparatus 10 includes a logic processor 14, which can be a general purpose microprocessor, such as an Intel Pentium or DEC Alpha,

of the type a personal computer or engineering workstation, a digital signal processor (DSP), such as a Texas Instruments TMS320 line, or a specialized CPU, such as a media processor, or a custom processing circuit. Apparatus 10 also includes a storage unit 18 which preferably includes random access memory. Because the algorithms used by the invention are not computationally intense, they require calculations on the order of less than one million instructions per second and can be performed by most modern personal computers.

FIG. 1 shows that in step 20, a portion of the original data is read into storage unit 18. The original data may represent, for example, sound that is recorded by sampling its amplitude periodically, with each sample using binary numbers to represent the magnitude of the sound at a particular time. Step 24 shows that the sample data is investigated sequentially by the logic processor to locate sample points that meet a predefined detection criteria. Such sample points indicate the existence of "local masking opportunities," because the detection criteria are such that a change in the value of the sample or a few samples at or near that point to embed auxiliary data will have little or no effect perceivable by the listener of the sound. The same detection criteria will be applied during data extraction to locate the hidden data.

Each point in the original data is preferably investigated to determine whether it represents a local masking opportunity. The criterion or criteria for determining local masking opportunities may entail not only the value of the point being investigated, but may also include the value of at least one nearby or neighboring point, or the relationship between the

nearby point and the point being investigated. The detection criteria can require, for example, that the point being investigated exceed a certain threshold value and/or that the point be a local maximum or peak. The criteria may include a requirement that a point subsequent to the point being investigated have a value that differs from the point being investigated by less that a prescribed amount, or have some other relationship to the point being investigated.

The sample data points can be considered to be plotted on a graph, for example with time on the x-axis and the magnitude of the sample on the y-axis. Thus, the series of data points can be considered as having a slope between any points, and the value of the slope can be part of the detection criteria. The criteria may specify, for example, that a slope defined by the point being investigated and a preceding point, exceed a particular value, or that the change in slope before and after the point not exceed a particular value. The criteria could include any combination of requirements. The threshold criteria can be changed to meet the needs of specific applications without departing from the concept of the invention.

In each case, no complex data transformation is required to mask the auxiliary data, so comparing a point to the detection criteria is relatively quick and inexpensive. Unlike prior art methods, which need to use distant points to convert the original data into the frequency domain to determine how to mask embedded data, the present invention can determines masking opportunities using only nearby or neighboring points, that is, points that are too close to use to determine useful frequency data. Nearby points including points that are next to the point being investigated or within a relatively small number of points, preferably less

than 50 and more preferably less than 20. The criterion can be as simple as determining whether the point exceeds a threshold.

Step 26 shows that when a point meeting the detection criteria is located, the value of a specified sample point or sample points near the local masking opportunity is changed to reflect the value of the auxiliary data to be embedded. Although the changed sample may be simply set to a particular value to signify the value of the embedded bit, the value to which the changed sample point is set typically depends not only upon the value of the auxiliary data to be embedded, but also upon the value of the point or points that were investigated to detect the local masking opportunity.

Thus, the point can be set so that it has a particular relationship to the point that was investigated. For example, the point may be set so that the change in slope signifies whether the embedded bit is a "1" or a "0," or so that a particular change in value from the point being investigated represents a "1" or a "0." When a point is set to a new value, it is important that the change does not prevent the original sample point from continuing to meet the detection criteria, because the same criteria that is used to locate places to embed data is used for locating points at which data is embedded. If the changed nearby point were to make the sample data point fail the detection criteria, the embedded auxiliary data will be lost because it will not be recognized when the detection criteria is applied to the combined data for retrieval of the auxiliary data. Alternatively, it is possible to merely encode the auxiliary bit as the least significant bit, or other, preferably low order, bit. The embedded bit is still

masked because the location of the embedded bit was chosen to represent a local masking opportunity, such as when the data is larger than a prescribed threshold.

Step 30 shows that the process is ended at step 32 if no additional auxiliary data needs to be embedded. Otherwise, step 34 shows that if there is additional data in memory, the search for local masking opportunities continues. Step 36 shows that if all data in memory has not yet been searched, additional data is read into memory. Skilled persons will recognize that some overlap of the data in memory is required to prevent missing local masking opportunities that occur at the beginning or end points of the data in memory.

FIG. 3 shows broadly the steps involved in carrying out a method of the invention to detect and extract the embedded auxiliary data. Because the same processor and memory that was used to embed the data can be used to extract the data, the steps of FIG. 3 will describe extracting data using the hardware components of FIG. 2. Step 50 shows that a portion of the original data is read into storage unit 18. Step 52 shows that logic processor 14 investigates each data point to determine the existence of a local masking opportunity. If a sample point meets the local masking opportunity criteria, step 54 shows that the embedded bit of auxiliary data is extracted, preferably by determining the relationship between the sample point and a specified nearby point, whose relationship determined whether a 1 or a 0 was embedded. Step 56 shows that if additional combined data is in the memory, the logic processor continues to investigate the remaining points with step 52. Step 58 shows that if all the data in memory has been investigated, but there is uninvestigated combined data in the data file,

additional data is read into memory in step 50. Step 60 shows that the process is ended when all the combined data has been investigated.

Although the examples herein describe original data that represents a series of sound magnitudes measured over time, each sample expressed as a group of bits, the original data could be any series of binary data associated into groups.

Two preferred embodiments are described briefly here, and in detail below, to demonstrate the flavor of this methodology. In the first embodiment, large, positive peaks are the detection criteria **120** and the auxiliary information is stored in the difference **130** between the peak and the next point. In the second embodiment, the detection criteria **140** are large, steep threshold crossings with minimal change in slope, and the auxiliary information **150** is carried in the change in slope.

The methodology is applicable to analog or digital data, even though the preferred embodiments use digital data. For example, analog data can be sampled at the Nyquist rate to produce digital data in which additional information is hidden. Then, the combined digital data can be returned to the analog domain by any existing method known in digital signal processing (DSP). The analog data now contains the embedded data, which can be decoded by using sampling. This is just one possible method to encode analog data with the above methodology.

The methodology is applicable to audio, speech, images, video or any other perceivable signal. With audio and speech, the original data could represent pressure versus time, magnitude versus frequency, or a specific frequency magnitude versus time. With images, the original data could represent gray code versus

space, separate or combined RGB or equivalent values versus space, or magnitude versus frequency. Video data encompasses the image data with an added dimension of time available.

Usually one of the detection criteria is a large threshold. With 16 bit audio, a threshold greater than 48 dB above the minimum value is desirable. This threshold allows the data to be changed with minimal perception due to masking. Masking is the psychological term that describes how one set of data covers the perception of other data. Specifically, the sensitivity of the sensory system decreases with increased input level, thus the small adjustment of an neighboring data point is masked by the large value of the threshold. In summary, the parameters of the detection criteria will determine the interaction between the data rate, process complexity and perceptual quality.

Embodiment 1

The first preferred embodiment is based upon hiding the auxiliary information in large peaks within the original data. In this embodiment, the auxiliary information is preferably broken into N bit words, with synchronization data placed between the words for better error recovery. The auxiliary information does not need to be grouped into N bit words with sync pulses between the words if robustness to noise or modified files is not needed.

FIG. 4 shows conceptually that the first embodiment detects a peak or local maximum and sets the value of the subsequent point in relation to the peak to indicate the value of the embedded bit.

FIG. 5 demonstrates the pseudocode in the form of a flowchart for the encoding process. First, the

original data is searched until a positive peak that lies above a large threshold, labeled thr, and has a relatively small decrease after the peak, labeled dS, is found. This process is demonstrated in boxes **200**, **210** and **220**. The detection criteria is checked in the order that is most computationally efficient. First points are sequentially checked to see if they represent a peak and are checked for exceeding the threshold only if they represent a peak. Second, when a desirable peak is found, the data point after the peak is adjusted according to a user defined bit depth, b, to carry the auxiliary information. Specifically, if it is the beginning of an auxiliary word, the synchronization code is embedded by adjusting the point after the peak, x[n+1], to be the peak, x[n], minus half of the maximum allowable change, dS/2, between the peak and the next point, as shown in boxes **242**, **230** and **250**. An auxiliary information bit of one is encoded by adjusting the point after the peak, x[n+1], to be equal to the peak, x[n], minus half the maximum change, dS/2, and plus the half the bit depth magnitude, 2^{b-1}, whereas an auxiliary information bit of zero is encoded by adjusting the point after the peak, x[n+1], to be equal to the peak, x[n], minus the sum of half the maximum change, dS/2, and half the bit depth magnitude, 2^{b-1}. This embedding of zeros and ones is shown in boxes **242**, **240**, **260**, **270** and **280**. The point after embedding the data can be skipped for efficiency as shown in box **290**. These steps are repeated until the auxiliary information has been hidden in the original data or the original data is finished.

FIG. 6 demonstrates the pseudocode in the form of a flowchart for the decoding process of the first preferred embodiment. First, the original data is searched until a positive peak that lies above a large

threshold, labeled thr, and has a relatively small decrease after the peak, labeled dS, is found. This process is demonstrated in boxes **300**, **310** and **320**. Second, when a desirable peak is found, the difference between the peak and the data point after the peak is measured to retrieve the auxiliary information. Specifically, if the peak minus the point after the peak, x[n]-x[n+1], is close to half of the maximum allowable change, dS/2, a new auxiliary word is beginning, as shown in boxes **330** and **350**. If the peak minus the point after the peak, x[n]-x[n+1], is approximately equal to half the maximum change, dS/2, minus half the bit depth magnitude, 2^{b-1}, an auxiliary bit of one is found. If this difference, x[n]-x[n+1], is close to the sum of half the maximum change, dS/2, and half the bit depth magnitude, 2^{b-1}, an auxiliary bit of zero is retrieved. This retrieving of zeros and ones is shown in boxes **340**, **360**, **370**, **380**, and **382**. Finally, the point after retrieving the data can be skipped for efficiency as shown in box **390**. These steps are repeated until the auxiliary information has been retrieved in the original data or the original data is finished.

There are three user-defined parameters, including threshold, thr; bit depth, b; and maximum allowable change after the slope, dS. The threshold is usually around 48 dB above the minimal quantization, as discussed above for 16 bit audio. The bit depth is an indication of the relative change to be made to the sample point to embed the data. Thus, the smaller the bit depth, the less disturbance of the original data, making the embedded data less perceptible to the listener, but less robust, that is, more susceptible to being lost to noise or attack. Minimal perception in 16 bit audio is found when bit depths are between 1 and

6 bits. However, higher bit depths can be used if one desires more robustness to noise in trade for more perceptual degradation. The maximum allowable change after the peak, dS, must be at least the desired bit depth magnitude, 2^b. On the one hand, one can gain better robustness to noise at the expense of more distortion, if dS is set to twice the bit depth magnitude, 2^{b+1}. On the other hand, if one desires to keep the threshold undetectable to statistical cryptoanalysis (labeled statistically invisible), dS should be set at 2^b, and b should be small, probably below 3 bits. If dS is not 2^b, one can use the discrepancy of the average difference between large positive peaks and their next points between embedded file and regular file data to determine if the file is embedded at all or modified. Finally, if dS is much greater than 2^b, the auxiliary information embedding rate will be increased, because more peaks will be found suitable for data embedding. Using the principles explained above, skilled persons will be able to set the user-defined parameters to values appropriate to the requirements of a particular application.

As discussed above, the large threshold causes this process to be non-perceivable due to masking. In addition, many data points satisfy the small difference between the peak and data point after the peak, because with a slope near 0 at the peak, the data is changing the least. This small difference means that the adjustment will be small as compared to the threshold.

The pseudocode is shown using a buffer with what appears to be look ahead capabilities (i.e. x[n+1]). This makes the process easier to explain and understand. However, the process is causal, as determined by replacing n+1 with k, and keeping track

of the last two points, x[k-1] and x[k-2]. Thus, it can be implemented with or without a buffer.
Finally, one can add more criteria to define the peak. For example, the peak extends for one more point each direction where x[n]>x[n-2] and x[n]>x[n+2], or the peak is a minimal sharpness, i.e. x[n]-x[n-1]>5. Both of these criteria produce better robustness to noise and less distortion since it will take more noise to move the location of the peak, although changes in the peak criteria affect the rate at which auxiliary data can be embedded.

The embedded data density and bit rate will vary with the original data and with the user-defined parameters. For example, bit rates of between 99 and 268 bits per second were achieved in CD quality audio data using a bit depth of 5 and a threshold of 5,000. Using a bit depth of 8 and maintaining a threshold at 5,000, an embedding rate of about 1,000 bits per second is expected. When the threshold is lowed to 2,000 at a bit depth of 8, data is embedded at 2,000 bits per second.

Embodiment 2

The second preferred embodiment hides the auxiliary information in threshold crossings which do not have a large change in slope. The method is more robust to noise changing the detected location. This occurs because it is less likely that noise changes the location of a threshold crossing as compared to a peak, since a threshold crossing usually has a slope larger than the slope at the peak, which, by definition, has a slope near zero. Testing with audio data has shown this embodiment to produce a lower embedded data rate and is perceivable at a lower bit depth, in trade for the robustness to noise. One will probably find the

optimal embodiment dependent upon application.

FIG. 7 shows conceptually that data is embedded by setting the slope after the threshold crossing in relation to the slope at the threshold crossing.

In FIG. 8, the pseudocode for hiding the auxiliary information using the second preferred embodiment is presented in the form of a flow chart. The process is as follows. First, the original data is searched until a positive, large, steep threshold (labeled thr) crossing with minimal change in slope (labeled dS) is found. This process is demonstrated in boxes **400**, **410** and **420**. Second, when the desirable threshold crossing is found, the data point after the threshold crossing is adjusted according to a user defined bit depth (b) to carry the auxiliary information in the change in slope. Note that the change in slope is defined as (x[n+1]-x[n])-(x[n]-x[n-1]), or equivalently as x[n+1]-2*x[n]+x[n-1]. Specifically, if it is the beginning of an auxiliary word, the synchronization code is embedded by adjusting the point after the threshold crossing, x[n+1], so that the change in slope is zero, as shown in boxes **442**, **430** and **450**. An auxiliary bit of one is encoded by adjusting the point after the threshold crossing, x[n+1], so that the change in slope is positive by an amount equal to half the bit depth magnitude, 2^{b-1}, whereas an auxiliary bit of zero is encoded by adjusting the point after the threshold crossing so that the change in slope is negative by an amount equal to half the bit depth magnitude, 2^{b-1}. This embedding of zeros and ones is shown in boxes **442**, **440**, **460**, **470** and **480**. The point after embedding the data can be skipped for efficiency as shown in box **490**. These steps are repeated until the auxiliary information has been hidden in the original data or the original data is finished.

FIG. 9 demonstrates the pseudocode in the form of a flowchart for the retrieving of the auxiliary information in the second preferred embodiment. First, the original data is searched until a positive, large, steep threshold (labeled thr) crossing with minimal change in slope (labeled dS), is found. This process is demonstrated in boxes **500**, **510** and **520**. Second, when a desirable threshold crossing is found, the change in slope around the threshold is measured to retrieve the auxiliary information. Again, the change in slope is defined as $(x[n+1]-x[n])-(x[n]-x[n-1])$, or equivalently as $x[n+1]-2*x[n]+x[n-1]$. Specifically, if the threshold crossing has almost zero change in slope, a new auxiliary word is begun, as shown in boxes **530** and **550**. If the threshold crossing has a positive change in slope approximately equal to half the bit depth magnitude, 2^{b-1}, an auxiliary bit of one is found. If the threshold crossing has a negative change in slope approximately equal to half the bit depth magnitude, 2^{b-1}, an auxiliary bit of zero is retrieved. This retrieving of zeros and ones is shown in boxes **540**, **560**, **570**, **580**, and **582**. Finally, the point after retrieving the data can be skipped for efficiency as shown in box **590**. These steps are repeated until the auxiliary information has been retrieved in the original data or the original data is finished.

As mentioned above, one does not want the embedding process to eliminate the embedded location from fulfilling the detection criteria. Specifically, in this embodiment, the pre-threshold change condition, $x[n]-x[n-1]>dS+2^{b-1}$, in the detection criteria of box **420** and **520** requires that the adjustment of the next data point does not bring the point back below the threshold. An alternative approach, is to ignore this condition and to set either the current or next point

(x[n] or x[n+1], respectively) to the threshold if the embedding process would cause the next point to move below the threshold, and ignore any data points that are equal to the threshold in both the encoding and decoding process. Interestingly, only when embedding a sync or 0 could the next point move below the threshold. The described embodiment is chosen so the process is causal, thus incorporating the known advantages of causal processes.

Once again, the large threshold and maximum allowable change in slope condition, dS, cause this process to be non-perceivable due to masking. The maximum allowable change in slope condition, dS, can have any value. A larger value allows a higher data rate with more perceivable distortion, whereas a smaller value produces minimal distortion with a lower data rate. Our preferred setting for 16 bit audio is equal to the bit depth magnitude, 2^b. Again, bit depths below 6 bits produce minimal distortion, but higher bit depths can be used for robustness to noise and attack.

Using a threshold of 2,000 and a bit depth of 5, a data rate of around 40-100 bits per second is expected, with an average of about 75 bits per second, for CD quality audio. At a bit depth of 8, the bit rate increases to an average of about 100 bits per second.

Options

The preferred embodiments have been described in detail above. However, there are many simple modifications that can be made to optimize the process for each use. Thus, these modifications and many similar ones produce a process that is equivalent to the one taught in this disclosure.

In some applications, a very simple embodiment

could use a simple threshold to determine a local masking opportunity and then encode the auxiliary data in the LSB of the point exceeding the threshold or of another point in the vicinity of the point exceeding the threshold. Such a variation is extremely simple, yet provides reduced perceptibility compared to prior art LSB schemes, because it uses masking. As with the other embodiments, one must ensure that changing the value does not remove the point for the detection criterion. In this case, one could simply skip embedding where the change brings the data below the threshold, and change the current value of the data point to the threshold so that the data point will be skipped in the decoding phase.

To increase the robustness of the invention to attack or noise, the following changes could be made. (Attack is defined as a person or machine trying to remove the auxiliary information from the combined signal without distorting the perception of the original data.)

Using a dynamic threshold can make it harder to remove the auxiliary information. An example dynamic threshold is an offset sinusoidal waveform. When using a dynamic threshold, dS should be small and close to 2^b so that the process does not change the distribution of the differences between neighboring points, i.e. be statistically invisible; thus, an attacker cannot use this data to find the threshold.

One can also use the statistical gaps when dS is larger than 2^b to find the threshold if the attack uses a DC shift. A DC shift is obviously a more potent attack for the second preferred embodiment than the first, but could affect the first preferred embodiment since threshold is one of the detection criteria.

The process could use more global definitions for

peaks and threshold crossings, for better robustness to noise. Specifically, a peak or threshold crossing definition that includes more points on each side.

The process can use any type of encrypting on the auxiliary information before embedding it to increase robustness to attack. The encrypting could be based upon the value of the original data at the insertion location to make it dynamic, and, thus, harder to discover.

Finally, the process can use any type of error correction in the auxiliary information to increase the robustness.

To increase the data rate, the following changes could be made. The auxiliary information does not need to be grouped into N bit words with sync pulses between the words if robustness to noise is not needed. In addition, negative going peaks and/or more thresholds can be used to increase bit rate. Finally, the process can use more than a binary system in adjusting the second bit to encode more information. However, the result is more likely to be perceivable or less robust to attack.

An interesting twist is to embed different auxiliary information on positive and negative peaks, and/or on various thresholds. In addition, with stereo files, you can code the channels separately or move between channels with consecutive points moving between left and right channels.

A change that could improve the perception is to move the data point after the embedded point towards the value of the embedded point if combining the auxiliary information causes a large value change in the embedded point.

Finally, as mentioned above, the data does not have to be relative to time. For example, the data

could represent magnitude versus frequency. In addition, the data could be viewed as magnitude of a specific frequency versus time. You could include all frequencies for an increased data rate. In other words, you could embed in the spectrum or spectrogram.

Example Utilizations

Below are included some example utilizations of the algorithm to aid in its understanding. This list is not complete, and only highlights the usefulness of the invention. The invention in its various forms is useful in any application in which it is desirable to embed auxiliary data into original data in a minimally perceptible or imperceptible manner.

The process can be used to embed copyright information. This information could include a code to determine if the data can be copied. Copying devices, such as CD writers, could include an inexpensive integrated circuit that could interpret to embedded data and prohibit copying.

In addition, author's or artist's name and affiliation can be embedded. In this utilization, the auxiliary information is small and would be repeated over and over with synchronization pulses between each duplication. Alternatively, the copy code could be embedded using embodiment 1, and the creator's name and affiliation using embodiment 2.

The invention can also be used to send additional information. This information could be transmitted in ASCII or ANSI with 8 bit "words" (not to be included with digital words being defined as 32 bits) and synchronization pulses between these words, if desired. The information could be a secret message, lyrics to the song, or a description of the artwork. For lyrics, this could be useful for kareoke machines and CD or DVD

players.

Digital Compression

The main problem with hiding data and digital compression (reducing bit rate not dynamic range) is that the process of hiding data is incompatible with compression. This incompatibility occurs since the goal of data hiding is to make the data minimally perceivable and the goal of compression is to remove minimally perceivable parts.

To this end, FIG. 10 demonstrates the process for data hiding, if at some point in the transmission process the data must be compressed. On the pre-transmission side, the auxiliary information is embedded in the non-compressed data using the described invented process, as shown in box **600**. Then, when the data needs to be compressed for transmission, the auxiliary information is retrieved via the described invention, and re-embedded in the compressed data with an appropriate scheme, as shown in box **610**.

On the post-transmission side, the auxiliary information is retrieved from the compressed data using the appropriate algorithm, the data is uncompressed, and the auxiliary information is hidden in the uncompressed data using the described invention, as shown in box **620**. Finally, when needed, the auxiliary information can be retrieved from the data using the described invention, as shown in **630**.

Apparatus

FIG. 11 shows that the invention typically comprises an apparatus **700** that includes a data reader **710** for reading original data **720** and auxiliary data **730**, a comparer **740**, that is, a circuit or device for comparing data points with known values or other data

points, and a data writer **750** for writing the combined data **760** to a permanent or temporary storage media.

As described above, FIG 2 demonstrates that the invented process can be implemented via logic processor and storage unit 18. FIG. 13 shows the implementation with a digital processor **800** and digital memory **810**. The digital processor **800** may be defined as the equivalent of a digital signal processor (DSP), general-purpose central processing unit (CPU), or a specialized CPU, including media processors. A likely DSP chip is one of the Texas Instruments TMS320 product line. A CPU could include one of Intel's Pentium line or Motorola/IBM's PowerPC product line. The design is straightforward for someone familiar with the state of the art given the pseudocode in Figs 5 through 9.

In addition, a person familiar with the state of the art could implement the process with analog and digital circuitry **900**, either separate or in an application specific integrated circuit (ASIC), as shown in Fig. 12. The analog and digital circuitry **900** could include any combination of the following devices: a digital-to-analog converter (D/A), comparators, sample-and-hold circuits, delay elements, analog-to-digital converter (A/D), and programmable logic controllers (PLC). Someone familiar with the state of the art given the previous description and pseudocode in FIGS 5 through 9 could easily design the circuit.

Conclusions, Ramifications and Scope

As the reader can see from the description above and determined from testing the process with audio, this process and apparatus of hiding auxiliary information within original data is non-perceivable and efficient. These advantages are mainly due to the invented process finding locations to hide the

auxiliary data without needing to transform the signal to the frequency domain where masking will block the perception of the auxiliary data.

The foregoing descriptions of the preferred embodiments of the invention have been presented to teach those skilled in the art how to best utilize the invention. Many modifications and variations are possible in light of the above teaching. For example, as discussed above, the peak criteria can be extended, the threshold may be dynamic, synchronization codes, error correcting codes, and encryption can be used, and any combination of peaks and threshold can be used jointly. To this end, the following claims define the scope and spirit of the invention.

Claims

1. A method of embedding auxiliary data into original data representing information to be presented to a user, the auxiliary data consisting of a series of binary digits and the original data consisting of a series of groups of binary digits, the binary digits of each group representing a numerical group value, the method comprising:

investigating groups in the series to locate a first group that represents a local masking opportunity, the local masking opportunity being determined by comparing the value of the first group with a predetermined value and with at least one nearby group value;

changing a group value of a second group in the vicinity of the first group to embed a binary digit of the auxiliary data in the original data, the local masking opportunity reducing the likelihood of a user perceiving the change of group value when the combined data is presented to the user; and

investigating additional groups in the series to locate additional local masking opportunities and changing corresponding group values in the vicinity of the additional local masking opportunities until all the auxiliary data is embedded in the original data or until no additional local masking opportunities are located.

2. The method of claim 1. in which the second group immediately follows the first group in the series, and in which changing the second group value includes setting the second group value to a first value to represent a bit of auxiliary data having a value of 1 or setting the second group value to a second value to represent auxiliary data having a value of 0, the first and second values being dependent on

the value of the corresponding first group value as well as on the value of the embedded auxiliary data bit.

3. The method of claim 2. in which investigating groups includes determining whether the first group represents a positive peak that lies above a predetermined threshold and whether the second group value differs from the first group value by less than a predetermined amount.

4. The method of claim 3. in which setting the second group value includes setting the second group to a value that differs from the first group value by less than said predetermined amount.

5. The method of claim 2. in which investigating groups includes determining whether:

the difference between the first group value and a group value before the first group value is within a predetermined amount of the difference between the first group value and a group value after the first group; and

the group value before the first group and the first group value is below a predetermined value and the group value after the first group is above the predetermined value.

6. The method of claim 5. in which changing a group value in the vicinity of the local masking opportunity to embed auxiliary data includes setting the value of a point after the point being investigated so that the difference represents a bit of auxiliary data having a value of 1 or 0.

7. The method of claim 1. in which original represents audio information.

8. The method of claim 7. in which the auxiliary data is used to prevent unauthorized copying of the original data.

9.

10. A data storage media including combined data having auxiliary data embedded therein by the process described in claim 1.

11. A method of retrieving auxiliary data into original data representing information to be presented to a user, the auxiliary data consisting of a series of binary digits and the original data consisting of a series of groups, each group consisting of a collection of binary digits, the binary digits of each group representing a numerical group value, the method comprising:

investigating group values in the series to locate a local masking opportunity, the existence of a local masking opportunity being determined by comparing the value of a group being investigated with a predetermined value and with at least one nearby group value;

determining from a group value in the vicinity of the group being investigated the value of a bit of embedded auxiliary data; and

locating additional local masking opportunities and determining the value of additional bits of embedded auxiliary data.

12. An apparatus for embedding auxiliary data into original data representing information to be presented to a user, the auxiliary data consisting of a series of binary digits and the original data consisting of a series of groups, each group consisting of a collection of binary digits, the binary digits of each group representing a numerical group value, the method comprising:

a comparer device for comparing a group value in the series with a predetermined value and with at

least one nearby group value to determine whether the group value represents a local masking opportunity;

data writer for storing the original data and embedded data onto a storage media, the data writer changing a group value of the original data in the vicinity of the local masking opportunity to embed auxiliary data in the original data to produce combined data, the local masking opportunity reducing the likelihood of perception of the changed value when the combined data is presented to the user;

the comparator determining additional local masking opportunities and the data writer changing corresponding group values in the vicinity of the additional local masking opportunities until all the auxiliary data is embedded in the original data or until no additional local masking opportunities are located.

13. A method of hiding auxiliary information within original data composed of a first series of points to create combined data composed of a second series of points, each point having a data value, the method comprising embedding by setting or retrieving by measuring the data value of a point in the original or combined data at a position in the first or second series of points determined by a detection criterion, the detection criterion including the presence of a predetermined relationship between neighboring points in the original or combined data, whereby the detection criteria ensures that the auxiliary data is minimally perceivable in the combined data.

14. The method of claim 13. wherein the detection criteria includes locating a point in the original or combined data that:

represents a positive peak that lies above a threshold value; and

is followed by a point having a value that varies by less than a predetermined amount from the value of the peak.

15. The method of claim 13. wherein the detection criteria includes determining the difference in values between a point representing a peak and a subsequent data point in the series.

16. The method of claim 15. wherein auxiliary data is hidden within original data so as to be minimally perceivable.

17. The method of claim 15. wherein auxiliary data is retrieved from the original data.

18. The method of claim 13. wherein neighboring points in the original or combined data define, together with neighboring points, lines having slopes and wherein the detection criteria includes a first line having a relatively large, positive, slope and crossing a threshold value and a second, neighboring line that has a slope that differs from that of the first line by than a predetermined value.

19. The method of claim 13. wherein neighboring points in the original or combined data define, together with neighboring points, lines having slopes and wherein the detection criteria includes the difference in slopes of neighboring lines, one of which crosses a predetermined threshold value.

20. The method of claim 13. in which the auxiliary data is encrypted, thereby increasing robustness of the embedded data to attack.

21. The method of claim 13. in which the auxiliary data uses error recovery codes, thereby increasing robustness of the embedded data to attack.

22. The method of claim 13. further comprising digitally compressing the original information, wherein the auxiliary information is

embedded into and retrieved from the non-compressed data and wherein the auxiliary information is also embedded into and retrieved from the non-compressed data, whereby the auxiliary information is not lost in the compression process.

23. The method of claim 22. wherein the transformation between the non-compressed and compressed data includes:

retrieving the auxiliary information;

compressing the combined information;

re-embedding the auxiliary information using an appropriate technique; and

transmitting the compressed information, whereby the compressed data has the auxiliary information embedded.

24. The method of claim 22. wherein the transformation between the non-compressed and compressed data includes:

retrieving the auxiliary information from the compressed information using an appropriate technique;

decompressing the compressed information; and

embedding the auxiliary information in the decompressed information, whereby the non-compressed data has the auxiliary information embedded.

25. An apparatus for hiding or retrieving auxiliary information within original data to produce combined data, comprising

a logic processor; and

a storage unit for storing information used to determine whether data points in the original or combined data meet a predetermined criterion or criteria;

whereby the logical processor searches the original data or combined data using a predefined detection criterion or criteria and embeds by setting or retrieve by measuring auxiliary data, the predefined

detection criterion or criteria using the relationship between neighboring points in the original or combined data.

26. The apparatus of claim 25. wherein the logic processor is digital processor and the storage unit is digital memory.

27. The apparatus of claim 25. wherein the logic processor and the storage unit consists of a combination of analog and digital circuitry

28. The apparatus of claim 25. wherein at least a portion of the auxiliary information is encrypted, thereby increasing robustness to attack.

29. The apparatus of claim 25. wherein at least a portion of the auxiliary information includes error correction codes.

30. A method of embedding data within original data, the original data being divided into a series of groups, each group in the series characterized by a numerical value, the method comprising:

investigating a first group to determine the presence of a local masking opportunity; and

if the first group indicated the presence of a local masking opportunity, embedding data by setting the value of the first group or setting the value at least one nearby group in accordance with the value of a portion of the auxiliary data, the local masking opportunity allowing data to be embedded with minimal perception by a user of the data.

31. The method of claim 30. in which determining whether the first group represents a local masking opportunity includes examining the value of the first group and its relationship to a value of at least one other group near the first group.

32. The method of claim 30. in which embedding data by setting the value of one of the

nearby groups includes setting a value so that it has a predefined relationship to the value of the first group.

33. The method of claim 30. in which embedding data by setting the value of one of the nearby groups includes setting a value dependent upon the value of the first group and on the value of the auxiliary data.

34. The method of claim 30. in which:

determining whether a first group represents a local masking opportunity includes determining whether the first group exceeds a threshold; and

embedding data by setting the value of the first group includes changing the value of one or more bits of the first group.

35. The method of claim 30. in which each group represents a sample of audio data.

36. The method of claim 30. in which each group represents a value of the same quantity measured at a different time.

37. The method of claim 30. in which nearby groups include groups within 50 groups sequentially of the first group.

38. A method of retrieving data embedded within combined data, the combined data being divided into a series of groups, each group in the series characterized by a numerical value, the method comprising:

determining whether a first group represents a local masking opportunity by examining the value of the first group and values of at least one group near the first group; and

if the first group represents a local masking opportunity, retrieving embedding data by measuring the value of one of the local groups.

39. The method of claim 30. in which each group represents a sample of audio data.

40. The method of claim 30. in which each group represents a value of the same quantity at a different time.

41. A method of preventing unauthorized copying of audio data files, the audio data file being divided into a series of samples, each sample in the series characterized by a numerical value, the method comprising:

determining whether a first sample represents a local masking opportunity by examining the value of the first sample and values of at least one other sample near the first sample;

if the first sample represents a local masking opportunity, embedding a portion of unauthorized copy prevention data by setting the value of one of the nearby groups in accordance with the value of a portion of the copy prevention data, the local masking opportunity allowing the portion of the unauthorized copy prevention data to be embedded with minimal perception by a user of the audio file;

locating additional local masking opportunities to embed additional unauthorized copy prevention data within the audio data file; and

upon attempting to copy the audio data, verifying whether the combined data includes copy prevention information and, if so, preventing a copy from being produced.

42. The method of 41. in which the unauthorized copy prevention data is distributed throughout the audio data file.

Abstract of the Disclosure

This patent application demonstrates possibilities to hide auxiliary information within original data such that the auxiliary information can neither be perceived by humans nor found by statistics, known as watermarking or steganography. One use of this hidden data is to copyright digital media, including audio, images or video, to stop this illegal duplication process. The invented process involves using detection criteria within the original data to adjust neighboring data to carry the auxiliary information. The detection criteria allow the process to find locations to embed the data where masking covers its perception without the transformation to the frequency domain. The algorithm can be adjusted according to the desired tradeoffs between perception, coding rate and robustness to attack. The main advantage of this algorithm is its efficiency, thus allowing it to be implemented at a much lower cost than competing algorithms.